Visualisation on a Shoestring: a low-cost approach for building visualisation components of industrial digital solutions

Giovanna Martínez-Arellano [1], Michael J. McNally[1], Jack C Chaplin[1], Zhengyang Ling[2], Duncan McFarlane[2], and Svetan Ratchev[1]

Abstract The rate of adoption of digital solutions in manufacturing environments remains low despite the benefits these can bring. This is particularly acute among industrial small and medium enterprises (SMEs), who typically do not have the confidence to adopt new technologies and for which cost and a lack of skills remain key barriers. Most digital solutions require some type of visualisation component, being a vital way to the interpret and use effectively the data. Data visualisation on its own provides an opportunity to bridge the gap of digitalisation in SMEs by providing them invaluable process insights in an efficient manner without requiring high levels of training or expertise. However, as with other digital technologies, software components such as data analytics and visualisation are commonly developed, deployed, and maintained by a third party, and SMEs lack the expertise to understand how to implement or change visualisations and how they can be applied in the manufacturing domain. The Digital Manufacturing on a Shoestring approach proposes using off-the-shelf components, both hardware and software, to develop low-cost digital solutions with minimal expert knowledge. The underlying Shoestring architecture enables the incremental connectivity of different solution components using a service-oriented approach. This paper introduces the implementation of visualisation-as-a-service, where the visual components of a digital solution is dynamically created by a set of reusable, configurable and modular elements. We also introduce the use of templates for the no-code creation of visual solutions, taking advantage of the re-usability of visual components across different digital solutions.

1 Introduction

The lack of productivity increase in UK businesses over the last 20 years has been partially attributed to the failure to adopt industrial digital technologies (IDT) such as the those based on the Internet of Things (IoT), data analytics, and machine learning (ML) [6]. In addition to the complexity of defining a clear digitalisation strategy, small and medium enterprises (SMEs) are particularly challenged by the cost and a

[1] Institute for Advanced Manufacturing, University of Nottingham,[2] Institute for Manufacturing, University of Cambridge

lack of available digital skills to develop and deploy digital solutions [10]. Digital solutions typically involve the development of various components such as sensing, data acquisition and storage, data analysis, machine learning, and graphic interface components for data input and data visualisation [3]. Applications such as condition and process monitoring, machine vision for quality inspection, predictive maintenance, job tracking, digital work instructions, which have been identified as some of the high priority solutions for manufacturing SMEs [15], require some or all of the aforementioned elements. Building these solutions can become a complex task, on the one hand due to the amount of technologies available for each of these elements, and on the other hand, due to the skills needed to program and integrate them.

Visualisation elements are a key component in most of these digital solutions. Companies are often more interested in quick but effective visualisation of data before going for more complex data analytics techniques [3]. Visualisation has been identified as an essential, low-cost, and ”simple” way to start understanding the benefits of digitalisation and allow companies to kick off their digitalisation journey [1]. However, there are still challenges when it comes to implementing visualisation in a manufacturing environment, particularly if done with a low cost approach. Firstly, going for a low-cost visualisation technology, such as open source visualisation libraries, requires programming skills which many SMEs can lack. Secondly, if the visualisation is a bespoke solution developed by a third party, they can be difficult for the SME to adapt, reconfigure, or expand according to the end-user needs and availability of data. Thirdly, even when flexible and customisable visualisation tools and dashboards are used, there is a level of experience required to choose the right visualisation element that will support the data interpretation and decision making. For visualisation to be easily implemented in a manufacturing environment, these solutions need to be implemented in a no-code manner, where the end-user can benefit from pre-existing solutions to easily put together, configure, deploy, and maintain a visualisation solution in the most cost-effective manner. The Digital Manufacturing on a Shoestring project aims to lower the cost and skill barriers associated with the development and deployment of digital solutions through the use of *off-the-shelf* non-industrial technologies and through the development of a service-oriented architecture that enables easy integration of all these elements, called *building blocks* in the Shoestring project. In this paper, we introduce the concept of visualisation-as-a-service with reusable visualisation building blocks and an approach to easily develop and deploy visualisation and graphic interfaces as part of a low-cost digital solution. By doing this, the end-user can build visualisation components which are suitable for the application at hand without any code. The rest of this paper is organised as follows: Section 2 provides a brief introduction to the role of visualisation in industry, current trends as well as existing limitations for the uptake of visualisation tools in SMEs. Section 3 introduces the concept of a visualisation building block and its encapsulation as a service within the Shoestring architecture. Section 4 introduces the Shoestring framework for building digital solutions, particularly for visualisation building blocks. Section 5 presents use cases and pilot programs where these building blocks have been utilised. Finally Section 6 presents a discussion on the advantages, current limitations and future work.

2 Related Work

2.1 Visualisation

Visualisation is a crucial component of many digital solutions, as it enables the effective interpretation of data, improving decision making. Current manufacturing environments use visual systems such as multi-coloured stack lights, dials, and LCD displays to support visual monitoring and management of processes. With the recent emergence of the Industrial Internet of Things (IIoT), there has been a trend towards the use of ubiquitous dashboards, making data available remotely as well as in the shop floor through mobile devices. Compared to established human-machine interface counterparts, dashboards can provide a more customisable interface, allowing access to deeper data analytics and providing a stepping stone for moving from traditional manufacturing approaches towards an Industry 4.0 approach [1]. Some commercially available tools such as Tulip[1] and Braincube[2] offer customisable production dashboards. However, as is the case with Tulip, both hardware and software components are vendor locked making the digital solution more difficult to integrate with digital solutions from other vendors. This is an important issue, as companies with no clear digitalisation strategies end up with isolated low-scale solutions. While larger corporations start to benefit from the advantages of increased digitalisation and visualisation, SMEs who typically cannot muster the capital to make large investments on a digital infrastructure, are left behind and at a disadvantage.

With the availability of cloud infrastructures, web-based visualisations are becoming the norm across most IIoT applications. An advantage of using a web based approach to visualisation is the large amount of open source web technologies available for developing graphic user interfaces and advanced data visualisations. Some of the low cost off-the-shelf tools available include Grafana, Python Dash, Shiny Dashboard, Bokeh, and GrapeJS. Grafana provides a no-code drag and drop approach to dashboard building, making it simple to configure different types of visualisations and to connect them to independent data sources. This tool is particularly aimed at the visualisation of time series data, although through the manual addition to the HTML code it is possible to include video and forms for data input. The dashboard can be exported or imported in the JSON format, making it easy to save and share as plain text. Python Dash, Shiny, and Bokeh are examples of libraries for the development of interactive dashboards. Compared to Grafana, these libraries provide a wider range of visualisation widgets and common HTML widgets such as buttons, slider inputs, and drop-downs, which allow for more interactive interfaces. Despite being free available technologies, the main challenge for the use of these is the level of skill that is needed to use them. Tools that are becoming very popular are those for the quick development of websites. GrapeJS, for example, allows the creation and styling of web pages through drag and drop and a CSS configuration

[1] https://tulip.co/

[2] https://braincube.com/

manager. The templates can then be exported, and deployed. GrapeJS is an effective solution for graphic interfaces that do not require any specialised data visualisations. In order to add data visualisation to any web page other technologies such as D3.js are needed. With the exception of Grafana, all these technologies require programming knowledge for developing and integrating them with other solution elements. Often these technologies would be used to develop bespoke solutions that frequently cannot be customised to adapt to different visualisation needs or to be extended into more complex visualisation tasks. Although open source technologies are key to developing low-cost solutions, there currently are no off-the-shelf low code building frameworks for open industrial visualisation solutions. Some efforts on the development of supporting tools for the design and development of these can be found in the literature [7, 8, 9], however these are typically technology specific.

2.2 Low-cost Digital Solutions for SME Manufacturers

For manufacturing digital solutions to be genuinely low cost, the costs of development and deployment also need to be low cost in addition to the cost of the technologies used [3]. The Digital Manufacturing on a Shoestring project has developed an approach to design, implement and deploy digital solutions, focusing in the following aspects [10, 11]:

1. Use of commercial off-the-shelf hardware and software
2. Low cost integration, operation and maintenance of solutions.
3. A systematic "building block" approach to combining different components.
4. Implementation of solutions through an *incremental architecture*.
5. Development of high-priority solutions for a majority of manufacturing SMEs.

As part of previous work done in the project [15], key priority areas common across most SMEs where digital solutions can provide huge advantage have been identified. From this set of digital solutions, what we refer as the *catalogue of digital solutions* has been drawn, and common building blocks among these solutions have been identified. The top ten solutions of this catalogue include:

- Job tracking
- Unified change management and issue reporting between design and production
- Digitised work instructions
- Capacity monitoring
- Analysis of product or customer demand
- Internal lead time monitoring
- Simulation of tools and processes for virtual process planning
- Cost modelling of disruptions and changes
- Digital job cards
- Process monitoring

Alongside the investigation and identification of these priority areas, the Shoestring project has focused on the development of a solution design approach that allows the systematic implementation of solutions and enabling the re-usability of components [3, 4]. The approach consists of three stages: (a) the identification of services through user questionnaires, (b) the specification of the necessary building blocks for each service using schematics and (c) the specification of the technologies. This process is streamlined through a online tool to simplify the development for end-users.

2.3 Visualisation in Low Cost Digital Solutions

The Shoestring design approach is oriented to the re-usability of components, both the services and building blocks within. As with hardware, visualisation is a common component across multiple priority digital solutions. Each of these solutions have common re-occurring data visualisation components. An example of this is the display of product and job related information. This information is required across multiple solutions such as job tracking, digitised work instructions, capacity monitoring, digital job cards, process monitoring, among others. In the case of graph-type visualisations, Gantt charts (see Figure 1) are a good way to visualise the change through time of a parameter or state. This visualisation can be used for job tracking, process/machine monitoring, capacity monitoring, to name a few. This led to the idea of building re-usable visualisation components that could be easily created and integrated into complete dashboards or web-based user interfaces across different digital solutions. The time and programming knowledge needed for developing these will be reduced, but offering these reusable components through the online tool will support the end-user to use the appropriate visualisation for the given task.

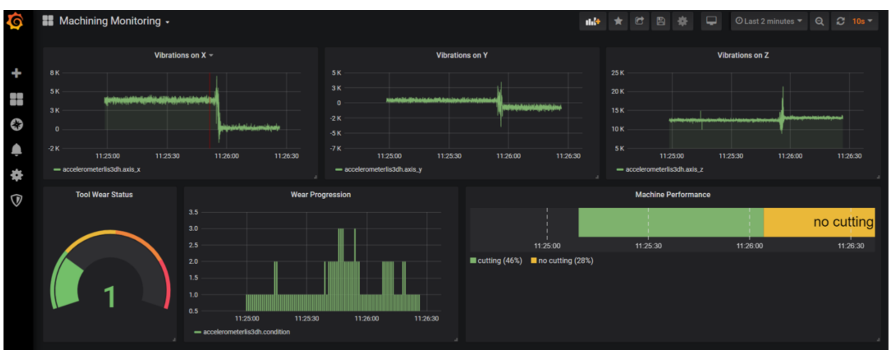


Fig. 1: Machining Dashboard which uses a Gantt chart visualisation (bottom right) to show the changes in cutting and no cutting periods of the machine, showing the current state with text and colour.

3 Shoestring Architecture and the Visualisation Building Block

3.1 Shoestring Architecture

The Shoestring architecture and associated design process are reported in more detail elsewhere and we provide a brief overview here. The architecture, which connects all re-usable components together, is based on the concept of building blocks and services that can be integrated through common communication standards [2] (Figure 2). A *service module* is an independent assembly of hardware and/or software that realises a fundamental digital manufacturing functionality. For example, a sensing service module would provide available measurements through a socket [3]. Service modules provide and consume data using a service layer communication

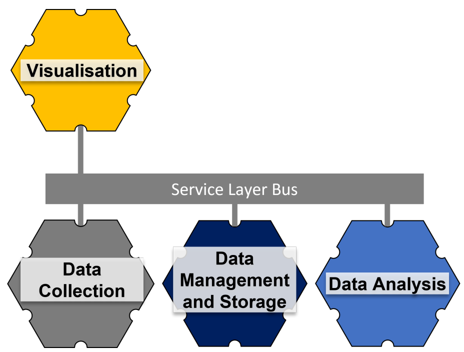
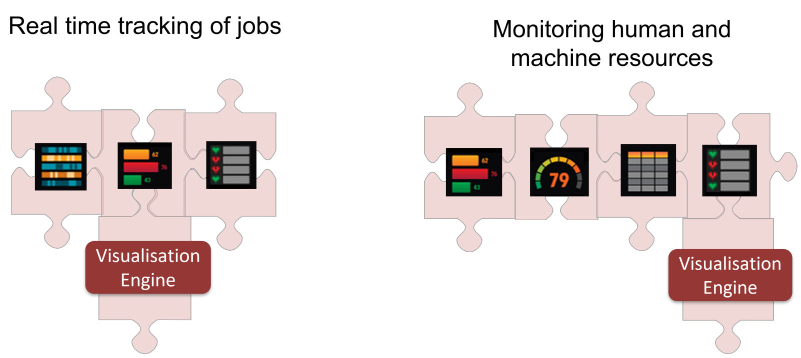


Fig. 2: Services connect through a common communication bus. This allows for the easy addition and removal of services to build a solution.

technology such as REST, OPC-UA, MQTT, or MTConnect [12, 13, 14]. Wrapping the different components of the solutions decouples the solutions so that they can be easily re-used as well as extended. The *building blocks* in the Shoestring architecture lie within the service module and provide a basic function. In the case of the sensing module, the sensor, the computing device, and the scripting language to read the data from the sensor and send it through a socket will all be building blocks that comprise the service. Building blocks are readily available hardware and software components that can be used to develop low-cost systems. Through this service oriented architecture it is possible to wrap both legacy systems and new technologies, allowing seamless connectivity among solution components and the adaptability of solutions across multiple production systems. Unlike other service oriented architectures [9, 16], the Shoestring approach is not layered, meaning a visualisation service can communicate directly to either a data storage or a data analytics service.

3.2 Visualisation in the Shoestring Architecture

The visualisation service wraps a set of visualisation building blocks to create a “complete” visualisation solution. Visualisation building blocks can be re-used across multiple digital solutions. Figure 3 shows an example of reuse across two visualisation solutions. The same bar chart building block can be used to visualise (a) the progress of jobs, and (b) the utilisation of a resource. The visualisation engine within the service module is independent of the visualisation building block. Engines such as R (Shiny), Dash, and Grafana as well as HTML pages can be used to integrate all building blocks into one visualisation solution. To achieve this, each widget or visualisation panel is encoded in the JJSON format, which can then be interpreted and rendered by the engine.



(a) Building blocks used when building a job tracking solution

(b) Building blocks used when building a resource monitoring solution.

Fig. 3: Visualisation building blocks can be re-used to build different solutions

Once all the building blocks within a visualisation solution are defined and integrated through the engine, these are wrapped as a service using ZeroMQ. This means that rather than having (for example) a Grafana panel directly querying a data source, the panel is connected to the service wrapper which then sends the request through ZeroMQ to (for example) a data storage service or data analytics service (Figure 4).

4 Automatic Configuration of Visualisation Components

4.1 Building Solutions using a No-code Approach

Implementing visualisation services using the Shoestring architecture shown in Section 3 can be complex for the inexperienced user as it requires the development of

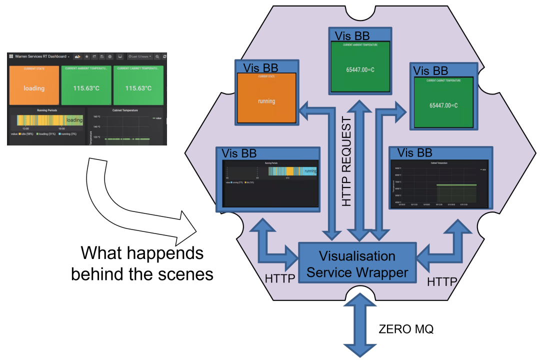


Fig. 4: The visualisation services consume data from other services such as the data storage and the analysis service. Within the visualisation service module, building blocks request data to be displayed through the visualisation service wrapper.

software components such as the visualisations themselves as well as the infrastructure for these to communicate to the service wrapper. In order for these to be easily adopted by SMEs, visualisation services and other other solution components need to be created in a no-code manner. For this reason, the Shoestring project has developed a framework referred to as the *Solution Configurator* [5] that allows users to drag and drop different solution components for both hardware and software. To achieve this, different visualisation and HTML widgets that are common across multiple digital solutions have been pre-defined using templates so that the user can easily configure them through the solution configurator (Figure 5). The solution configurator then ensures that once the solution is defined and finalised by the user, all the necessary service wrappers are automatically created including the visualisation service wrapper. Depending on the type of visualisation engine that has been selected by the user, all the necessary code for the building blocks to be integrated will be generated together with instructions for the deployment of the code and installation of any necessary libraries. This way a fully automated solution generation is achieved. A pre-defined layout of visualisation widgets is used during generation but a more experienced user can further customise the visualisation widgets if needed.

4.2 Templates for Reusable Visualisation Components

In addition to the common visualisation widgets used for building dashboards, other graphical interface needs have been identified based on the solutions catalogue. Digital solutions not only need graphical data display but also text based data output and data input. For this reason, a set of templates for HTML forms and common information display components have been designed. Figure 6 shows some of the general templates for two digital solutions; job tracking and digital work instructions.

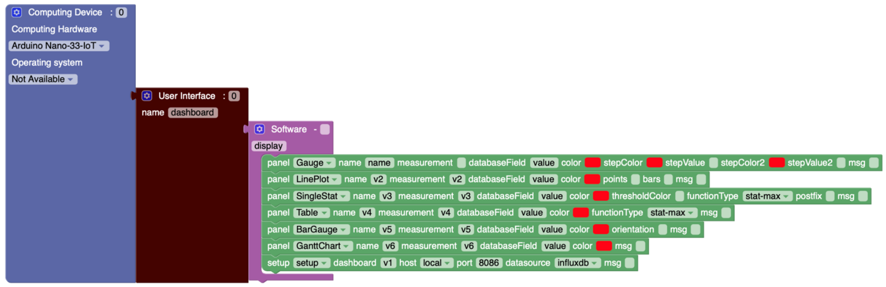
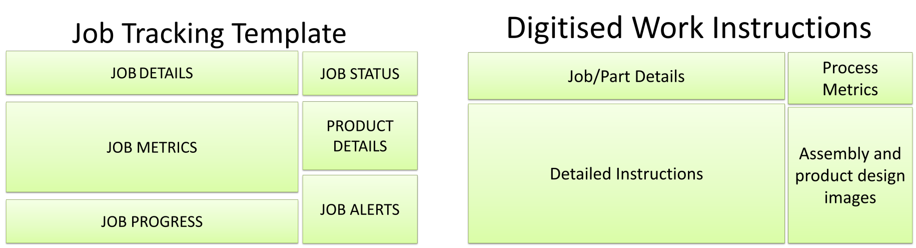


Fig. 5: The Shoestring Solution Configurator allows users to drag and drop different visualisation building blocks and configure them. When the solution is ready, the solution configurator produces a visualisation component in the JSON format that can then be rendered by different visualisation engines.

Within these *base templates*, there are some re-occurring components such as the product details or job details. In addition to the layout base template, templates for these re-occurring components are defined to enable reusability across solutions. This template approach is implemented using Django. Depending on user selection, specific HTML components can be rendered using the base template. Figure 7 shows an example of a visualisation solution generated through templates, where the product detail area is rendered through a pre-defined product component.



(a) (b)

Fig. 6: General layout templates for different HTML-based visualisation solutions

5 Visualisation Pilot Study

To validate the concept of visualisation-as-a-service and demonstrate its integration with other service modules, a pilot study was carried out in conjunction with Warren Services, one of the Digital Manufacturing on a Shoestring project industrial partners. Sensor data was made available via an API, which was wrapped as a data

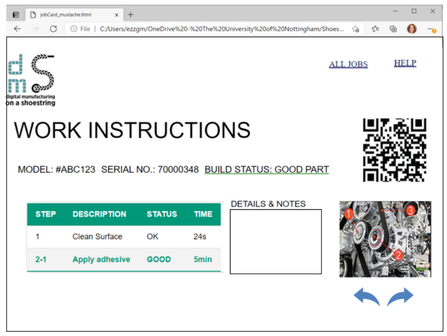


Fig. 7: An example of a visualisation component generated through HTML templates and Django.

collection service. A data analysis module was developed as a Flask (Python) application to post-process the data before being presented in a visualisation dashboard, making various metrics available for display through HTTP requests. Finally, two visualisation services were created, one using Grafana and another one using R to demonstrate how the architecture enables the incremental incorporation of multiple services, in this case two visualisation services (Figure 9). In both cases, each of the visualisation building blocks within each service perform a request of data or metrics through the service wrapper to the data analysis service. When a request is received by the data analysis service through its service wrapper, this service then requests the necessary data to the data collection service in order to calculate the metrics. Once the data is received and calculations performed, the data analysis service sends the metrics to the visualisation service. As it can be observed in Figure 8, some services such as the data analytics service works as a client as well as a server service module. In the case of the visualisation service, the module is mainly implemented as a client.

Figure 8 shows the final architecture that was implemented, using ZeroMQ as the communication technology. ZeroMQ was chosen as it is a lightweight messaging kernel that can be used across multiple platforms. At the stage of the development of the pilot, the solution configurator was at an early development stage, so these visualisation components were developed manually, although the creation of these supported the development of the JSON templates which are now integrated in the solution configurator. The pilot also enabled the validation of the communication architecture, which is able to now wrap and integrate any service during solution generation.

6 Discussion and Future Work

In this paper the conceptualisation and implementation of visualisation-as-a-service is presented following the low cost Shoestring approach to digital manufacturing. By

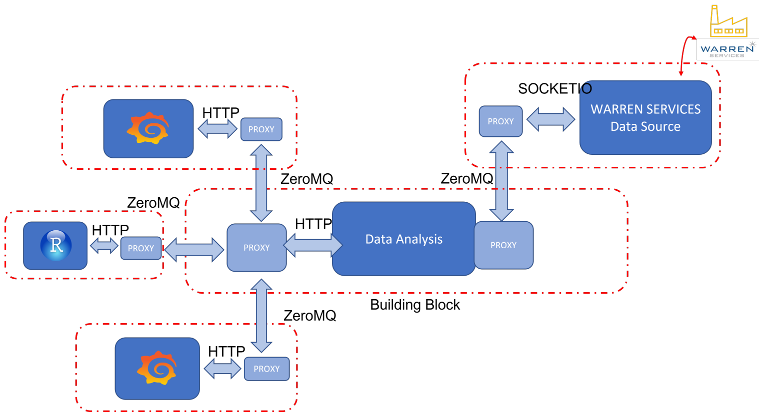
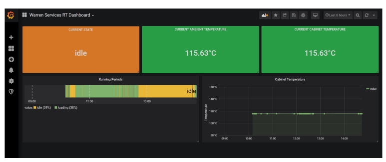
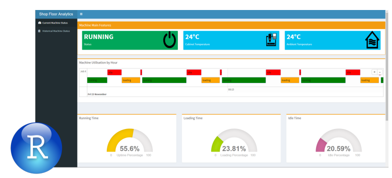


Fig. 8: Solution implemented for the Warren Services pilot study using two versions of the visualisation service, one using Grafana and another using R. Communication between service modules is implemented using ZeroMQ.



(a) Visualisation service for the Warren Services pilot study using Grafana engine.



(b) Visualisation service for the Warren Services pilot study using R engine.

Fig. 9: Visualisation services implemented in Grafana and R for the Warren Services pilot study. The service architecture facilitates the easy integration of multiple visualisation services to the same digital solution. In this case, both visualisation services consume data from the same data analysis service.

modularising visualisation components, it is easier to incrementally build visualisations and re-use these across multiple digital solutions. Modularisation is achieved through a building block and service architecture which decouples the visualisation technology from the rest of the solution components and services. In addition to the concept of visualisation-as-a-service, a drag and drop approach for the creation of no-code open source visualisations is presented and the use of templates for the flexible creation of HTML-based manufacturing user interfaces is proposed. Although this approach already lowers the barrier for the integration of visualisations by manufacturing practitioners, there is still a challenge on deciding which visualisations are most appropriate for the application at hand; which ones can deliver the highest value for better decision making. Recent studies have shown how visualisations' "pre-attentive attributes" allow users to quickly identify differences in data without much cognitive processing [17]. Future work will include the development of pre-defined solutions made available through the solution configurator, providing the user with a selection of the most suitable visualisation components for a given application. In addition, a higher level of configuration, particularly for HTML-based solutions will be added to the configurator for greater flexibility.

References

1. Allen, L., Atkinson, J., Jayasundara, D., Cordiner, J., Moghadam, P.Z.: Data Visualisation for Industry 4.0: A Stepping-stone Toward a Digital Future, Bridging the Gap Between Academia and Industry. Patterns. **2**(5) (2021).
2. Hawkridge, G., Perez Hernandez, M., de Silva, L., Terrazas, G., Tlegenov, Y., McFarlane, D., Thorne, A. Tying Together Solutions for Digital Manufacturing: Assessment of Connectivity Technologies & Approaches. In 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 1383—1387 (2019).
3. Hawkridge, G., Mukherjee, A., McFarlane, D., Tlegenov, Y., Parlikad, A. K., Reyner, N. J., Thorne, A.: Monitoring on a Shoestring: Low Cost Solutions for Digital Manufacturing. Annual Reviews in Control. **52**, 374–391 (2021).
4. Hawkridge, G., McFarlane, D., Kaiser, J., de Silva, L., Terrazas, G. Designing Shoestring Solutions: An Approach for Designing Low-cost Digital Solutions for Manufacturing. submitted to SOHOMA 2021.
5. Ling, Z., de Silva, L., Hawkridge, G., McFarlane, D., Martínez-Arellano, G., Schönfuß, B., Thorne, A. A Visual Environment for Developing Affordable Digital Manufacturing Solutions for SMEs, submitted to SOHOMA 2021.
6. Maier, J. et al. Made Smarter – Review (2017) https://www.gov.uk/government/publications/made-smarter-review. Cited 09 July 2021
7. Logre, I., Mosser, S., Collet, P., Riveill, M. Sensor Data Visualisation: A Composition-Based Approach to Support Domain Variability. European Conference on Modelling Foundations and Applications (ECMFA 2014), July 2014.
8. Vázquez-Ingelmo, A., García-Peñalvo, F.J., Therón, R. Towards a Technological Ecosystem to Provide Information Dashboards as a Service: A Dynamic Proposal for Supplying Dashboards Adapted to Specific Scenarios. Appl. Sci.**11**, 3249 (2021).
9. Lin, H., Chiu, Y., Liao, W., Chang, R. Service-Oriented Architecture for Intelligent Management with Data Analytics and Visualization, 2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA), 73-78 (2019).
10. McFarlane, D., Ratchev, S., Thorne, A., Parlikad, A.K., de Silva, L., Schönfuß, B., Hawkridge, G., Terrazas, G., Tlegenov, Y.: Digital Manufacturing on a Shoestring: Low Cost Digital Solutions for SMEs. Studies in Computational Intelligence. **853**, 40 – 51 (2020).
11. McFarlane, D., Ratchev, S., de Silva, L., Hawkridge, G., Schönfuß, B., Terrazas, G. Digitalisation for SME Manufacturers: A Framework and a Low Cost Approach. submitted to SOHOMA 2021.
12. https://mqtt.org/, last accessed on 30/08/21.
13. https://www.mtconnect.org/, last accessed on 30/08/21.
14. https://opcfoundation.org/about/opc-technologies/opc-ua/, last accessed on 30/08/21.
15. Schönfuß, B., McFarlane, D., Athanassopoulou, N., Salter, L., de Silva, L., Ratchev, S. : Prioritising Low cost Digital Solutions Required by Manufacturing SMEs: A Shoestring Approach. International Workshop on Service Orientaiton in Holonic and Multi-agent Manufacturing. 290 – 300 (2019).
16. Pivoto, D.G.S., de Almeida, L.F.F., Righi, R. da R., Rodrigues, J.J.P.C., Lugli, A.B., Alberti, A.M., Cyber-physical systems architectures for industrial internet of things applications in Industry 4.0: A literature review, Journal of Manufacturing Systems, **58**(A), 176–192 (2021).
17. Nimbarte AD, Smith N, Gopalakrishnan B. Human Factors Evaluation of Energy Visualization Dashboards. Ergonomics in Design, (2021).